Journal of Southeast University (English Edition) Vol. 22, No. 3, pp. 370 – 374 Sept. 2006 ISSN 1003—7985

Using ontology semantics to improve text documents clustering

Luo Na[1,2] Zuo Wanli[1] Yuan Fuyu[1] Zhang Jingbo[2] Zhang Huijie[2]

([1] College of Computer Science and Technology, Jilin University, Changchun 130012, China)

([2] School of Computer Science, Northeast Normal University, Changchun 130024, China)

Abstract: In order to improve the clustering results and select in the results, the ontology semantic is combined with document clustering. A new document clustering algorithm based WordNet in the phrase of document processing is proposed. First, every word vector by new entities is extended after the documents are represented by tf-idf. Then the feature extracting algorithm is applied for the documents. Finally, the algorithm of ontology aggregation clustering (OAC) is proposed to improve the result of document clustering. Experiments are based on the data set of Reuters 20 News Group, and experimental results are compared with the results obtained by mutual information(MI). The conclusion draws that the proposed algorithm of document clustering based on ontology is better than the other existed clustering algorithms such as MNB, CLUTO, co-clustering, etc.

Key words: ontology; text clustering; lexicon; WordNet

With the volume of knowledge and information available to computer users increasing at an ever accelerating rate, the need for an effective mechanism to organize not only information, but also knowledge becomes critically important. Document clustering techniques have been employed frequently to support the organization and retrieval of information[1]. One current problem of information retrieval systems is that it is not readily possible to automatically extract meaning from the relevant results of a query in order to support the user in the search process. One main reason for this is that the web is initially designed to direct human use[2] and thus documents do not provide machine readable semantic annotations.

The concept of the semantic web proposed by Berners-Lee[3] outlined the idea of using machine-processable semantics assigned to each available information resource, which started a new evolution of the world wide web. This approach should provide the possibility to sorting and structuring information in order to access and retrieve content more precisely. However, there are currently only a few web pages that provide semantic annotations, so we decided to use the already available external resources (in our case, ontologies) to assign meaning to documents in relation to a given query. The idea is to disambiguate their content similar to a user who is searching for information. At present, users have to navigate among many documents to select the relevant documents that they really require, because current retrieval systems do not provide such semantic information.

Received 2006-04-18.

Foundation items: The National Natural Science Foundation of China (No. 60373099), the Natural Science Foundation for Young Scholars of Northeast Normal University (No. 20061005).

Biographies: Luo Na (1980—), female, graduate; Zuo Wanli (corresponding author), male, doctor, professor, wanli@jlu.edu.cn.

1 Ontolgoies and Lexicons

An ontology is a formal specification of a conceptualization of a domain of interest. It specifies a set of constraints, which declare what should necessarily hold in any possible world. The intention is to build a complete world model for describing the semantics of information exchange. Especially in the area of artificial intelligence, ontologies are being used in order to facilitate knowledge sharing and reuse. Ontologies and lexicons are the core elements of our method.

Definition 1 Ontological layer: An ontology O: $=\{C, P, H_{c,p}, \mathrm{ROOT}\}$, which consists of ① A set of classes C (also means a set of concepts); ② Properties P, one class C may have many properties; ③ A class hierarchy H_c: Classes are taxonomically related by the directed, acyclic, transitive, reflexive relation H, ($H_c(C_1, C_2)$ means that C_1 is a sub-class of C_2); ④ A property hierarchy H_p: defined analogously to H_c; ⑤ A top class ROOT $\in C$, for all $c \in C$ it holds: $H(C, \mathrm{ROOT})$; ⑥ The ontological layer defines the concepts for preprocessing and selection of relevant views onto the set of texts. Here, we use, roughly, correspondence to the basic structures used in the famous WordNet[4].

Definition 2 Lexicon for an ontology: A lexicon for an ontology O is a set of terms. It is a tuple Lex: $=(S_c, \mathrm{Ref}_C)$ consisting of a set of S_c, whose elements are called signs for concepts(symbols) and a relation Ref_C

$\subseteq S_c \times C$ called lexical reference for concepts, where $(C, C) \in \mathrm{Ref}_C$ holds for all $c \in C \cap S_c$. Based on Ref_C, for $s \in S_c$, we define $\mathrm{Ref}_C(s) := \{c \in C \mid (s, c) \in \mathrm{Ref}_C\}$.

The definition allows $(n: m)$-relations between lexical entries and ontological entities; that is, a lexical entry may refer to several classes or properties and one class or property may be referenced by several lexical entries. Examples for the lexical layer could be {"Airplane", "Aeroplane", "Plane"} or {"beef", "pork"}.

2 Our Method

Both clustering and classification may benefit from the integration of prior external class knowledge, which reflects specific classification concepts or organization goals. The incorporation of domain knowledge into clustering can be applied in several phases. Fig. 1 illustrates three main stages at which ontology can be employed.

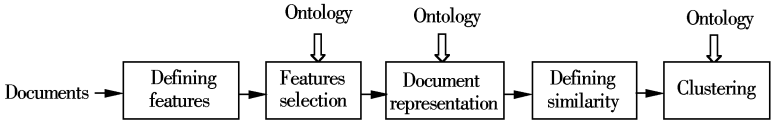

Fig. 1 Stages of document clustering using ontology

2.1 Document preprocessing

Documents may be represented by a wide range of different feature descriptions. Here, we use the vector space model (VSM), as described in the work of Salton et al. (1975), in which a document is represented as a vector or "bag of words"; i. e., by the "words" it contains their frequency, regardless of their order. The basic idea of VSM is using a vector to represent documents, as $(W_1, W_2, \ldots, W_n)$, where W_i is the weight of the i-th feature term. Generally, we select letters, words or phrases as feature terms. Experiments have shown that using terms as feature terms is better than letters and phrases. So we represent documents as a vector in vector space. The immediate drawback of this basic approach for document preprocessing is that if a document has many features in vector space, the clustering task will have to work in high-dimensional space where each word is seen as a potential attribute for a text. Empirical and mathematical analysis, however, has shown that clustering is inefficient in high-dimensional spaces and is difficult in theory. The technique we adopt in reducing the dimensions of clustering is based on the following steps.

1) Document representation

Let us first consider documents to be bags of terms. Let $\mathrm{tf}(d, t)$ be the absolute frequency of term $t \in T$ in document $d \in D$, where D is the set of documents and $T = \{t_1, t_2, \ldots, t_m\}$ is the set all different terms occurring in D. Term vectors $\boldsymbol{t}_d = \{\mathrm{tf}(d, t_1), \ldots, \mathrm{tf}(d, t_m)\}$. Here, we extend each term vector $\boldsymbol{t}_d$ by new entries for WordNet concepts c appearing in the document set. Thus, the vector $\boldsymbol{t}_d$ is replaced by the concatenation of $\boldsymbol{t}_d$ and $\boldsymbol{c}_d$, where $\boldsymbol{c}_d := (\mathrm{cf}(d, c_1), \ldots, \mathrm{cf}(d, c_l))$ is the concept vector with $l = |C|$ and $\mathrm{cf}(d, c)$ denotes the frequency that a concept $c \in C$ appears in a document d as indicated by applying the reference relation Ref_C to all terms in the document d, where $\mathrm{cf}(d, c) := \mathrm{tf}(d, \{t \in T \mid c \in \mathrm{Ref}_C(t)\})$.

2) Stopword removal, stemming and pruning

Stopwords are words which are considered as non-descriptive within a "bag of words" approach. Following common practice, we remove stopwords from T, using a standard list with 571 stopwords. We process our text documents using the Porter stemmer introduced in Ref. [5].

3) Weighting

Weights are assigned to give an indication of the importance of a word. The most trivial weight is the word-frequency. However, more sophisticated methods can provide better results. Throughout this work, we use the information retrieval measure tf-idf in selecting terms.

4) Tag the corpus

After the steps above we tag the corpus in order to find concepts and examine the effect of clustering. Here the concepts are produced not only by terms but also the term arrays are produced by suffix arrays. The experiment shows that tagging naïve word sense disambiguation can help to improve clustering results.

2.2 Feature selection

The amount of words constituting texts is quite huge, and the dimension expressing vector space of texts is great, in the tens of thousands. So we need to reduce the dimensionality of data. There are two reasons for doing that. First, improve running speed to increase efficiency. Secondly, the significances of all the words for text classification are not the same. Some general words appearing in all kinds of documents render little contribution for classification, but the words with proportions in special classes is large while the

proportions in other classes have little contribution for classification. To improve the precision of classification, for each class, we should remove the words which do not possess strong expressive power, and select the set of feature items of each class.

Algorithm 1 Feature extraction

Input: The set of classes and words in each class.

Output: The set of feature items.

Initially, the set of feature items includes the words in all classes.

Let $\{C_i\}_{i=1}^{m}$ denote the set of categories in the target space. The information gain of term t is defined to be

$$G(t) = -\sum_{i=1}^{m} P_r(c_i)\log P_r(c_i) + P_r(t)\sum_{i=1}^{m} P_r(c_i \mid t)\log P_r(c_i \mid t) + P_r(\bar{t}) \cdot \sum_{i=1}^{m} P_r(c_i \mid \bar{t})\log P_r(c_i \mid \bar{t}) \quad (1)$$

For all the words in the class, we compute the IG values and order them.

Selecting certain amounts of words as feature items offers no proper solution to the question of how many feature items we should select. The general method is providing initial values first (the initial value should be several thousand), then ensuring optimum value by experimental testing and statistical results.

For all training words in each class, we reduce the vector dimension to simplify the expression of vector by selecting feature items.

2.3 Integrating ontology into clustering

A word in different contexts may contain different concepts, thus a word may be placed in different synsets, from different hypernym trees. For example (see Fig. 2), to look up the hypernym of the word orange with the color concept, the left hypernym tree should be followed[6]. The hypernym for orange with the color concept is color, but with the fruit concept it is edible fruit. Based on these studies we choose to classify documents using the hypernyms (the superordinate words), the hyponyms (the subordinate words) and the relevant glosses. We use the hypernymy relation because it describes the superordinate word of our search words (words that are more generic), and thus separates one meaning from another. Instead of using only the superordinate hyperonym of the word, we extract all hyperonyms until we reach the primitive of the word. It means that we separate a category from another where they intersect. We also decide to use the hyponymy relation, because with this relation all subordinated words related to a word are included. Both of these relations describe the restricting context. We also choose to use the glosses (human readable descriptions of the words) included in the WordNet ontology, because they give a deeper description of the synsets elements of words that are frequently used in this specific semantic context.

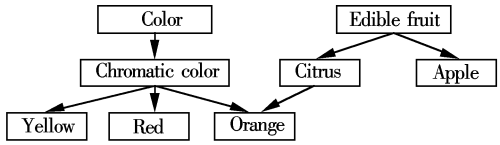

Fig. 2 An example of hierarchically semantic relationships among words

From Fig. 2, we can see that the principal idea is that if a term like "orange" appears, one does not only represent the document by the concept corresponding to "orange", but also by the concepts corresponding to "color" and "edible fruit" etc.

The following procedure realizes this idea by adding to the concept frequency of higher level concepts in a document d the frequencies that their subconcepts appear. The vectors we consider are of the form $\boldsymbol{t}_d := \{\mathrm{tf}(d, t_1), \ldots, \mathrm{tf}(d, t_m), \mathrm{cf}(d, c_1), \ldots, \mathrm{cf}(d, c_n)\}$. Then the frequencies of the concept vector part are updated in the following way: For all $c \in C$, replace $\mathrm{cf}(d, c)$ by $\mathrm{cf}'(d, c) := \sum_{b \in H(c, r)} \mathrm{cf}(d, b)$, where

$$H(c, r) := \{c' \mid \exists c_1, \ldots, c_i \in C: c' \prec c_1 \prec \ldots \prec c_i = c, 0 \leqslant i \leqslant r\} \quad (2)$$

After the document vectors are assigned to their respective WordNet subconcepts[7], we use the ontology aggregation clustering algorithm to fine-tune our results. We use the number of classes obtained from the ontology for the number of clusters k and start the clustering process using the ontology subconcepts as initial cluster centers.

The ontology aggregation clustering algorithm computes the cluster centers. Each object is assigned to the closest cluster center. Then the cluster centers are recalculated as the average of the document vectors assigned to the cluster. This process is repeated until there are no more changes to the cluster centers. The pseudo code description of ontology aggregation clustering algorithm is as follows.

Algorithm 2 Ontology aggregation clustering (OAC)

Input: Document vectors and the number of document n, cluster number k, the set of concepts c.

Output: The cluster in which the document is.

Initialize the k clusters,

Use WordNet, then let c be the set of centers of all clusters,

Repeat

For $i = 1$ to n do {

Based on the average value of objects in a cluster, classify the object with the closest cluster again}

For $j = 1$ to k do {

Calculate new cluster center using average of document vectors assigned to the cluster}

Continue until no more reallocations of documents occur.

3 Experiments and Results

In our experiments we use Reuters 20 News Groups data and a number of datasets from the CLUTO toolkit[8]. These datasets provide a good representation of different characteristics: number of documents ranges from 204 to 19 949, number of terms from 5 832 to 43 586, number of classes from 3 to 20, and balance from 0.036 to 0.998.

In the experiments we vary the different strategies for plain term vector representation and for vector representations using ontology as elaborated above. Here we use MI(mutual information) to analyze the results of clustering, and the definition of normalized mutual information(NMI) using geometrical method, as given in Ref. [9]. In practice, we use a sample estimate

$$\mathrm{NMI} = \frac{\sum_{h,l} n_{h,l} \log\left(\frac{n n_{h,l}}{n_h n_l}\right)}{\sqrt{\left(\sum_h n_h \log\frac{n_h}{n}\right)\left(\sum_l n_l \log\frac{n_l}{n}\right)}} \quad (3)$$

where n_h is the number of data samples in class h, n_l the number of samples in cluster l and $n_{h,l}$ the number of data samples in cluster l. The NMI value is one where clustering results perfectly match the external category labels and are close to 0 for a random partitioning. This is a better measure than purity or entropy, which is biased towards high k solutions[9−10]. In our experiments, we use NMI as the evaluation criterion. Tab. 1

Tab. 1 NMI results on Reuters 20 News Groups datasets

k	10	20	30	40
MBM	0.18 ± 0.05	0.19 ± 0.03	0.17 ± 0.02	0.18 ± 0.03
MM	0.52 ± 0.02	0.54 ± 0.04	0.54 ± 0.04	0.56 ± 0.02
CLUTO	0.55 ± 0.02	0.58 ± 0.01	0.58 ± 0.01	0.57 ± 0.01
Co-clustering	0.36 ± 0.01	0.46 ± 0.01	0.50 ± 0.01	0.51 ± 0.01
mvMF-clustering	0.57 ± 0.03	0.59 ± 0.02	0.57 ± 0.01	0.56 ± 0.01
Ontology-clustering	0.64 ± 0.02	0.66 ± 0.03	0.62 ± 0.01	0.61 ± 0.01

shows the results on Reuters 20 News Groups, from which it can obviously be seen that using MI, the ontology-based clustering method outperforms other methods.

4 Conclusion

In this paper, we discuss a way of incorporating ontology—WordNet into a representation for text document clustering in order to improve clustering results. We have performed evaluations on the Reuters data set. It shows that the proposed algorithm of document clustering based on ontology is better than the other existing clustering algorithms such as MNB, CLUTO, co-clustering, etc.

References

[1] Kim H J, Lee S G. A semi-supervised document clustering technique for information and organization [A]. In: *Proc of the Ninth International Conference on Information and Knowledge Management* [C]. McLean, Virginia, 2002. 159 − 168.

[2] Brusilovsky P. Methods and techniques of adaptive hypermedia [J]. *User Modeling and User Adapted Interaction*, 1996, **6**(2, 3): 87 − 129.

[3] Berners-Lee T, Hendler J, Lassila O. The semantic web[J]. *Scientific American*, 2001, **184**(5): 34 − 43.

[4] Abdelali Ahmed, Cowie James, Farwell David, et al. Cross-language information retrieval using ontology [A]. In: *Proc of Traitment Automatique des Languages Naturelles* [C]. Batz-sur-Mer, France, 2003. 236 − 248.

[5] Porter M F. An algorithm for suffix stripping [J]. *Program*, 1980, **14**(3): 130 − 137.

[6] Gruber T. A translation approach to portable ontology specifications [J]. *An International Journal of Knowledge Acquisition for Knowledge-Based Systems*, 1993, **5**(2): 62 − 69.

[7] Miller G. WordNet: a lexical database for English [J]. *Communications of the Association for Computing Machinery*, 1995, **38**(11): 39 − 41.

[8] Karypis G, Zhao Y. Evaluation of hierarchical clustering algorithms for document datasets [A]. In: *Proc of the International Conference on Information and Knowledge Management*[C]. New York, 2002. 515 − 524.

[9] Strehl A, Ghosh J. Cluster ensembles—a knowledge reuse framework for combining partitions [J]. *Journal of Machine Learning Research*, 2002, **3**: 583 − 617.

[10] Strehl A, Ghosh J, Mooney R J. Impact of similarity measures on web-page clustering [A]. In: *Proc of AAAI Workshop on AI for Web Search* [C]. Austin, Texas, 2000. 58 − 64.

使用本体语义提高文本聚类

罗　娜[1,2]　左万利[1]　袁福宇[1]　张靖波[2]　张慧杰[2]

([1] 吉林大学计算机科学与技术学院,长春 130012)
([2] 东北师范大学计算机学院,长春 130024)

摘要:为了提高聚类结果和允许在结果中进行选择,将本体语义与文档聚类相结合,在文档处理过程中提出了基于 WordNet 的新的文档聚类算法. 首先通过 tf-idf 对文档进行了表示,为了将 WordNet 的概念出现在文档集合中,通过新的实体对每一个单词向量进行扩展. 其次,运用特征提取算法对文档进行特征提取. 最后提出了本体集合聚类算法用以提高文本的聚类效果. 实验构建在 Reuters 20 新闻组的数据基础上,应用互信息作为试验结果的比较. 结果表明:与已经存在的一些算法如 MNB,CLUTO,co-clustering 等相比,基于本体的聚类算法在文本聚类上有很明显的提高.

关键词:本体;文本聚类;词典;WordNet
中图分类号:TP181